N95- 17601
32006
p. 8

Mission Operations Centers (MOCs): Integrating Key Spacecraft Ground Data System Components

Randy Harbaugh
National Aeronautics and Space Administration
Goddard Space Flight Center
Greenbelt, Maryland, U.S.A.

Donna Szakal
Computer Sciences Corporation
Laurel, Maryland, U.S.A.

ABSTRACT

In an environment characterized by decreasing budgets, limited system development time, and user needs for increased capabilities, the Mission Operations Division (MOD) at the National Aeronautics and Space Administration Goddard Space Flight Center initiated a new, cost-effective concept in developing its spacecraft ground data systems: the Mission Operations Center (MOC). In the MOC approach, key components are integrated into a comprehensive and cohesive spacecraft planning, monitoring, command, and control system with a single, state-of-the-art graphical user interface. The MOD is currently implementing MOCs, which feature a common, reusable, and extendable system architecture, to support the X-Ray Timing Explorer (XTE), Tropical Rainfall Measuring Mission (TRMM), and Advanced Composition Explorer (ACE) missions.

As a result of the MOC approach, mission operations are integrated, and users can, with a single system, perform real-time health and safety monitoring, real-time command and control, real-time attitude processing, real-time and predictive graphical spacecraft monitoring, trend analysis, mission planning and scheduling, command generation and management, network scheduling, guide star selection, and (using an expert system) spacecraft monitoring and fault isolation. The MOD is also implementing its test and training simulators under the new MOC management structure.

This paper describes the MOC concept, the management approaches used in developing MOC systems, the technologies employed and the development process improvement initiatives applied in implementing MOC systems, and the expected benefits to both the user and the mission project in using the MOC approach.

INTRODUCTION

The National Aeronautics and Space Administration (NASA) Goddard Space Flight Center (GSFC) Mission Operations Division (MOD), in partnership with the Computer Sciences Corporation (CSC) Control Systems Technology Group (CSTG), developed the Mission Operations Center (MOC) concept to improve the MOD's spacecraft ground data systems. The focus of this effort was to enhance system operability and increase functionality while lowering development and operational costs and shortening development time.

Four key advances within and outside the MOD arena contributed to the development and refinement of the MOC concept: reengineering of the MOD mission operations concept, restructuring of management to a mission-oriented structure, industry development of enabling technologies, and application of improvements in system development processes.

Reengineering of the MOD mission operations concept provided the framework for developing the MOC concept. Restructuring from a multimission to a mission-oriented management organization provided the vehicle for efficiently and effectively implementing the concept. Enabling technologies such as powerful workstations and industry standards

PAGE 1280

PRECEDING PAGE BLANK NOT FILMED

contributed to the feasibility of the concept. Improved system development processes in all life-cycle phases contributed to the cost-effectiveness of the concept.

DEFINING THE MOC CONCEPT

Driven by user demands for mission-unique systems with improved operability and increased functionality, mission profiles with accelerated spacecraft schedules, and NASA budgets in steady decline, the MOD reengineered its overall mission operations concept. This activity viewed mission operations from an MOD-wide level, with the goals of maximizing the operations that a single user can perform while minimizing system development time and reducing operational and development costs. The MOC concept makes significant strides toward achieving these goals.

Before the MOC, the MOD developed ground data systems and conducted mission operations in host-based, multimission environments supported by separate, independent branch organizations. For example, the Control Center Systems Branch (CCSB) (GSFC Code 511) developed Payload Operations Control Centers and the Spacecraft Control Programs Branch (GSFC Code 514) developed mission planning and command management systems. As a result of the reengineering activity, which encompassed the operational functionality of all of the MOD's independent systems, the MOC system was defined as an integrated, comprehensive, mission-unique system with a single user interface and the capabilities necessary to support the MOD's mission operations.

With a MOC system, the user can, from a single workstation, perform traditional mission operations including real-time spacecraft health and safety monitoring, real-time spacecraft command and control, trend analysis, mission planning, command generation and management, and network scheduling as well as newly added operations such as mission operations planning and scheduling, real-time and predictive graphical spacecraft monitoring, real-time attitude processing, guide star selection, and spacecraft subsystem monitoring and fault isolation .

The broadened view of the MOD's mission operations, free from the past organizationally induced partitions of functionality, enables comprehensive system engineering that considers only the technical aspects of the MOC system definition. The resulting MOC system eliminates redundant capabilities within the MOD; eliminates or simplifies interfaces to and within the MOD; and allows for cost-effective, systemwide solutions. Because of these improvements, a MOC system can be developed in less time and at lower cost than the traditional, independent ground data system implementations.

MANAGING MOC DEVELOPMENT

The concept of developing an integrated MOC ground data system naturally led to the concept of an integrated, mission-oriented management structure. To provide the vehicle for efficiently and effectively implementing each MOC system, both the MOD and CSC restructured their management organizations to create a single, mission-oriented MOC management team. Recognizing the potential for improving coordination and communication between the mission's MOC system and the mission's standalone test and training simulator [traditionally developed under the Simulations and Compatibility Test Branch (GSFC Code 515)], the MOD and CSC placed development of the simulator under the MOC management structure.

The resulting mission-oriented MOC organization is headed by a system manager and is supported by a MOC system engineer; managers of the major MOC components and the simulator; and knowledgeable, technical component experts. This approach retains the expertise of the traditional organizations and, for the first time, combines the functionality of the MOD ground data systems previously developed by independent organizations under a single MOD-level system manager.

The major advantages of this management approach over the traditional approach include consolidation of mission budgets; closer coordination of system capabilities and schedules; integration of a major portion of the mission ground system earlier in the ground system life cycle; provision for a single point of contact to the mission projects and users; and improvements in communication, coordination, and cooperation among the experts from the various ground data systems. Clearly, these advantages could only be realized when the management authority, responsibility, and control rested in the hands of a single, system manager whose primary focus was to manage development of the MOC system and the simulator.

Consolidating mission budgets under a single manager with the requisite responsibility and authority simplifies the planning of projected budgets and the reporting of actual spending on a per-mission basis. Further, single-manager responsibility for most of the MOC components results in increased flexibility in assigning resources among the components that need it. With the MOC approach, timely, system- and component-level budget decisions can be made within the MOC organization from a balanced and informed viewpoint.

Closely coordinating component development schedules and the capabilities to be implemented according to the consolidated MOC master schedule significantly improves the readiness of a major portion of the mission's total ground system. In the traditional approach, one ground data system's capabilities and schedules were usually developed with limited insight into the needs of the other ground data systems. This lack of close coordination sometimes resulted in the need for additional temporary software to simulate missing capabilities and delayed mission ground system testing of these capabilities. In the MOC approach, simulator and component schedules and capabilities are closely coordinated so that each MOC component fully supports the others and the MOC and simulator systems fully support each other at the scheduled time. This level of coordination significantly reduces time spent waiting for independent ground data systems to get synchronized in support of mission ground system testing. Although planning a MOC development schedule is slightly more time consuming and complex than in the traditional approach, monitoring projected and actual schedules is much quicker and easier because there is one composite schedule.

Integrating major portions of the mission's ground system during the development phase of the life cycle, rather than during the integration and test phase, significantly reduces mission ground system interface, integration, and end-to-end test time. The schedules and capabilities of the MOC system components are not only coordinated, but the major efforts of integrating and testing them and also testing the integrated MOC system with the simulator are accomplished by the development team before delivery of either system to GSFC. In the traditional approach, this integration and testing occurred after delivery of each of the independent ground data systems, when interface problems are difficult to isolate and repair quickly. The extensive, advanced planning of the MOC master schedule, which considers the project's test needs, coupled with the development team's expertise in integrating and testing the MOC system, improves the overall quality and readiness of the ground system earlier than previously possible.

Because the MOC organization provides a single point of contact, the MOD speaks with a unified voice to the mission project and MOC system users. Traditionally, a mission project had to communicate with each of the MOD branch organizations, an inefficient and time-consuming process. Also, users had to communicate with the developers from each of the MOD independent ground data systems to convey and receive information. The MOC approach ensures a direct, timely, and consistent flow of information from the MOC team to the mission project and the users. For example, with a MOC system, there is a single set of comprehensive, formal reviews (e.g., a single system requirements review, preliminary design review, and critical design review) to attend and critique; there are fewer documents to review and approve than with the traditional approach (e.g., a single comprehensive requirements specification rather than multiple ones); and, as an added benefit, the resources needed to prepare, present, and maintain these formal reviews and documents are reduced.

Improved communication, coordination, and cooperation among the technical experts from the various ground data systems ensures the timely development of robust, cost-effective MOC systems. The single, cohesive MOC team shares a common focus and a common goal: the successful implementation of the MOC system. The team makes decisions to support this goal, relinquishing conflicting demands and diverse approaches from the originating organizations in favor of a unified management and technical approach.

The MOC system engineer and component experts regularly share their expertise and insight with each other. Cross-checks of understandings highlight discrepancies early, allowing them to be solved when resolution is less costly. For example, on the X-Ray Timing Explorer (XTE) MOC, discrepancies existed in early mission documentation describing the telecommand packet checksum calculation. Because the simulator experts on the team knew how the flight software performed this calculation, they were able to resolve the problem quickly and with no cost impact. Typically, with the limited cross-check of understandings between simulator and control center system experts in the past, a problem such as this would not have been found until actual ground system integration testing with the spacecraft or the simulator, when problem repair is more costly.

The MOC team has also used their broadened view of the system to identify and implement more robust technical solutions. For example, the traditional simulator, control center, and command management systems each used different approaches and different data base software to process the mission's project data base. For the single, integrated MOC system, the team has identified and implemented a more rigorous relational data base solution with increased functionality over any of the traditional systems.

Working as a cooperative team, the MOC component experts have identified and eliminated redundancy among the components, reducing the amount of software that must be developed, tested, and maintained. Traditional capabilities, as well as new functional capabilities, are available earlier. For example, the selection of a single user interface means that the time traditionally spent developing and maintaining multiple user interfaces can be spent enhancing the functionality of the selected user interface. For the integrated MOC system, the team has also eliminated the formal interface between the control center and command management systems. Significant savings have been realized in eliminating the formal definition, negotiation, control, integration, and testing of this interface because these efforts are now performed within the MOC organization. Traditionally, several separate MOD and mission project organizations needed to be involved.

The most important challenges in defining the MOC management approach were to develop a mission-oriented organization that retained the expertise of the various components of the MOC system and to minimize the risks to the success of the mission while implementing the new technical and management approaches.

The MOC management approach capitalizes on the use of technical and management expertise from each of the ground data systems. The depth of knowledge provided by these component experts, coupled with the breadth of knowledge of the system engineer, is essential to the success of any MOC implementation. Management risks are minimized because component experts are part of the MOC team and because the best practices from the originating MOD branch organizations have been selected and implemented. Techniques such as the use of multimission working groups and matrices of expertise have been expanded to encompass the full MOC functionality. These techniques, successfully demonstrated in the traditional organizations, make possible high levels of software reuse across and within mission implementations. In the control center area, for example, software reuse levels of over 70 percent are regularly achieved. Technical risks are minimized because the selected MOC architecture is a natural extension of the in-place, highly successful system architecture described below. Use of these management and technical strategies minimizes the risk of the overall MOC concept.

To further reduce the risks, the MOD initiated a pilot project in October 1992, selecting the XTE mission as the first MOC system implementation. The MOD, CSC, and the mission project have closely monitored the progress of this MOC via various technical reviews and regular management reviews. By June 1993, the

XTE MOC pilot project showed such early promise and enthusiastic user support that the MOD reassigned the TRMM mission, which started development as separate control center, command management, and simulator ground data systems, as an integrated MOC system implementation and a standalone simulator under the new MOC management structure.

IMPLEMENTING THE MOC

MOC Architecture

The availability of enabling technologies such as powerful workstations and networks, distributed processing, commercial off-the-shelf (COTS) products, and industry standards contributed to the feasibility of the MOC concept. The effectiveness of their use was successfully demonstrated in the MOD's CCSB-developed Transportable Payload Operations Control Center (TPOCC) system philosophy and architecture. The MOC concept extends the use of TPOCC to cover broader functionality.

The TPOCC architecture is based on the use of industry standards, COTS components, custom reusable components, and distributed processing using client/server technology. It features interconnected hardware that provides systemwide access to data and distributed processing that is flexible and transparent to the user. It supports the dedicated use of a workstation for isolated functions or the use of a single workstation for multiple functions. It also allows single functions to be spread across multiple processors to provide needed levels of processing and data throughput. The state-of-the-art graphical user interface, which features a windowing environment, significantly increases system operability.

The TPOCC architecture is designed to be evolutionary in that new technology can be inserted into the basic system framework without disrupting the overall architectural approach. This extendable architecture easily supports integration of independently developed components that follow its fundamental precepts.

Each MOC system, which uses TPOCC's hardware architecture approach, is sized to meet its mission's data and operational needs and consists of a network of inexpensive, heterogeneous COTS workstations, X-terminals, and front-end processors (i.e., single-board computers). The architecture reflects a commitment to industry standards such as VME, Ethernet, RS-232, RS-422, and SCSI. For the MOC, a RAID array, optical disk, CD ROM, and 3-D graphics devices are added to the basic TPOCC architecture to support the broader functionality of a MOC.

The MOC software architecture approach, like TPOCC's, consists of distributed processing using client/server technology, adherence to open system communications standards, extensive use of COTS products, and implementation of reusable custom code. Most of the MOC software is written in C or C++ and is designed to be independent of the hardware, thus making it easily portable to other platforms.

All MOC software components are implemented following open system communications standards such as the Transmission Control Protocol/Internet Protocol (TCP/IP), external data representation (XDR), and network file system (NFS). The commercial standards for the MOC's graphical user interface include X-window and Open Software Foundation's Motif software. The use of industry standards facilitates incorporation of COTS products, generic systems, and independently built components without impacting the overall software architecture.

The MOC system is flexible and extendable. It supports, from a single workstation, the functionality previously dispersed among many minicomputers and mainframe systems, thus increasing the number of operations that a single user can perform. The MOC system reduces operational costs because multiple, independent systems are consolidated; workstations replace more expensive minicomputers and mainframe systems, and computer operators are no longer needed to support multimission computer facilities.

Process Improvements

Application of improved processes in the requirements, design, implementation, integration, and test phases of system development contributed to the cost-effectiveness of the MOC concept. In an atmosphere of continuous process improvement, the MOC development teams have applied several improvement initiatives to the development of the MOC systems.

The mission MOC teams have improved the process of defining the MOC system requirements. During the requirements definition phase, joint developer and user teams, sometimes referred to as Joint Application Development (JAD) teams, define the requirements. Using the JAD approach, users familiar with the specific mission requirements and operational needs and developers familiar with existing software capabilities are able to quickly identify mission-unique needs. The JAD team uses an existing set of requirements from other, similar missions as a base for defining the new mission's requirements. This approach results in the timely definition of requirements because the JAD team, rather than starting from scratch, simply analyzes the baseline and makes additions or deletions as appropriate. This approach also maximizes the reuse of existing software, limiting detailed requirements analysis to mission-unique areas. The Advanced Composition Explorer (ACE) MOC JAD team is using this approach with the XTE MOC requirements providing the basis for requirements discussions.

The ACE MOC JAD team is also piloting the concept of users and developers jointly documenting requirements rather than each group independently writing and cross-referencing separate, configuration-controlled documents. The team documents requirements on-line using the Requirements Generation System (RGS) data base tool. This approach is also expected to save considerable time and effort.

The mission MOC teams have also implemented improvements in the design process. During the design phase, extensive technical exchange meetings are held both within a specific mission MOC (i.e., cross-function) and across the MOCs of other missions (i.e., cross-mission). Each MOC's system engineer and component experts regularly hold cross-function technical exchange meetings to design portions of the software so that they can be used by multiple components, thus maximizing reuse within a mission MOC. In addition, the MOC system engineers and component experts regularly hold cross-mission technical exchange meetings to design specific components into generic and mission-unique building blocks, thus maximizing reuse across MOC missions (i.e., generic component software is designed with mission-unique "hooks"). This MOC design approach results in a comprehensive, cohesive system design that eliminates organizationally induced walls between functional components.

During the implementation phase, the mission MOC teams' strict adherence to system development standards and use of a standard user interface permits multiple components (i.e., multiple portions of the system) to be developed concurrently. Although this is not a new process, its implementation during MOC system development is essential. The mission MOC teams have also expanded the use of advanced COTS software development tools such as SoftBench, Branch Validator, and Purify to assist them in writing and debugging software.

The implementation of these development improvements allows the mission MOC teams to capitalize on three aspects of software reuse: reusing existing custom-built and generic software components; designing custom software with new functionality for future reuse; and integrating existing, standalone generic systems and COTS products. Each MOC team works with users to define mission requirements that maximize the reuse of existing custom-built and generic components (e.g., existing mission software, TPOCC generic software) while still meeting each mission's unique needs. Sharing requirements expertise across each mission allows the MOC teams to design custom code for future reusability because generic components are identified and developed to permit mission-unique extensions. Each MOC's design also integrates COTS products (e.g., ORACLE) and standalone generic systems such as the Generic Spacecraft Analyst Assistant (GenSAA) and the Generic Trend Analysis System (GTAS). The use of these techniques reduces the amount of new code needed while increasing functionality. For example, approximately 50 percent of the first MOC's code and over 80 percent of the second MOC's code consists of reusable components (not including the integrated generic systems or COTS products). This percentage is expected to increase as additional MOC capabilities are implemented for future reuse and spacecraft standards continue to be formulated and implemented.

The mission MOC teams have also instituted improvements in the integration phase. One of the major challenges for any MOC system is the integration of the many components that it comprises. Successful integration of a MOC system is a special and complex problem. The complexity of integrating "externally" developed components (i.e., components developed by other organizations), for example, encouraged definition of a formal integration procedure. This procedure includes requirements for extensive planning, preparation, and monitoring of the integration activity. For example, for components developed within the MOC organization, the MOC system engineer and component managers require demonstrations of, and explicit documentation about, each developer's software before that software is integrated with the total MOC system. In addition to these improvements, for the first time, the mission's test and training simulator has been collocated with the MOC system in the development environment, substantially improving the developers' ability to test the integrated MOC system. The MOC and simulator developers' ability to extensively exercise their systems before they are delivered to GSFC significantly improves the quality and robustness of each system.

During the test phase, the test process has been improved by combining traditionally separate system, acceptance, and user test teams into a single test team (independent of the development team) and by moving this level of testing from the traditional postdelivery timeframe into the predelivery timeframe. The combined, concurrent testing by this team reduces overall MOC system test time while increasing testing effectiveness. When the test team finds problems that must be repaired before the system is deemed ready for operational use, the development team corrects the problems. This extensive, independent, predelivery testing of the integrated MOC system reduces the amount of time necessary for mission-level ground system interface and integration testing because not only have some of the traditional interfaces been eliminated, but also a major portion of the overall ground system has been tested.

BENEFITS OF THE MOC APPROACH

The MOC approach provides major benefits to its users. Probably the most important of these benefits is the integration of mission operations with mission specialists collocated in the MOC's office-like, workstation environment. Traditional, host-based systems located in various multimission computer rooms required that users be able to operate several independent systems. On each of these systems, a user could perform only one operation from each terminal, requiring that user to monitor up to three or four terminals at a time, depending on the number of simultaneous operations to be performed. The MOC's mission-oriented, integrated system, with a windowing environment and distributed processing, allows the user to perform and monitor multiple operations from a single workstation, a vast improvement over traditional systems.

A second important benefit to the user is the MOC's state-of-the-art, standardized graphical user interface that provides the same "look and feel" across all components of the MOC. In addition to traditional tabular data displays, this interface supports graphical data representations such as plots, bars, dials, pie charts, and timelines, enabling users to rapidly distinguish anomalous situations. Menus and input panels are intuitive to operate, and, with only one consistent user interface to learn, user system training as well as cross-component training is simplified. This improved system operability, coupled with the increased functionality provided by a MOC system, provides the user with all the tools needed to perform operational duties.

The MOC approach provides many benefits to the mission project. The MOC management structure provides the mission project with a single point of contact for a major part of the developing mission ground system. This improves and simplifies communication both to and from the mission project.

Another major benefit to the mission project is that MOC systems, as opposed to traditional implementations, are less costly and achieve operational readiness in a shorter period of time. With fewer system interfaces, operational and system development complexity and associated costs are reduced. MOC approaches such as elimination of redundant code among components, extensive software reuse, integration of COTS products and existing generic systems, and commitment to expanding the library of reusable custom components by designing for future reuse are recognized approaches that reduce costs. The MOC systems contain more capability and higher stability early in the development cycle because of the extensive reuse of existing, tested software and COTS products.

SUMMARY

The MOC approach to ground systems development makes great strides toward integrating the MOD's mission operations. This approach significantly increases the number of operations that a single user can perform simultaneously, substantially improves system capability and operability, and simplifies user training, while reducing operations and development costs and shortening development time.

Only 2 years since its inception, the MOC concept has realized its initial goals. As the MOC approaches continue to mature, and as more functionality is incorporated into its systems, the benefits to mission projects and the user community are expected to grow.

REFERENCES

Koslosky J., Mahmot R.., Mandl D., and Stratton W. (June 1990). *Transportable Payload Operations Control Center,* Proceedings of European Space Agency Symposium: Ground Data Systems for Spacecraft Control (pp. 653-658).

Beach E., Giancola P., Gibson S., Mahmot R.. (November 1992). *Customizing Graphical User Interface Technology for Spacecraft Control Centers,* Proceedings of SPACEOPS '92 (pp. 485-490).

Carlton D., Vaules, Jr. D., Mandl, D. (November 1992). *GSMS and Space Views: Advanced Spacecraft Monitoring Tools,* Proceedings of SPACEOPS '92 (pp. 375-380).